|  |  |  |  |
| --- | --- | --- | --- |
| **Course Name:** | **Digital Image Processing** | **Semester:** | **VII** |
| **Date of Performance:** | **November 4, 2022** | **Batch No:** | **DIP2** |
| **Faculty Name:** | **Prof. Gopal Gupta** | **Roll No:** | **1912052** |
| **Faculty Sign & Date:** |  | **Grade/Marks:** |  |

**Experiment No: 6**

**Title: To study image filtering and Histogram stretching**

|  |
| --- |
| **Aim and Objective of the Experiment:** |
| To study the process of image filtering and histogram stretching |

|  |
| --- |
| **COs to be achieved:** |
| 1. **Understand fundamental theory and models of image processing** |

|  |
| --- |
| **Theory:** |
| There are two methods of enhancing contrast. The first one is called Histogram stretching that increase contrast. The second one is called Histogram equalization that enhance contrast and it has been discussed in our tutorial of histogram equalization.  Before we will discuss the histogram stretching to increase contrast, we will briefly define contrast.  Contrast  Contrast is the difference between maximum and minimum pixel intensity.  Consider this image.  stretching  The histogram of this image is shown below.  stretching  Now we calculate contrast from this image.  Contrast = 225.  Now we will increase the contrast of the image.  Increasing the contrast of the image  The formula for stretching the histogram of the image to increase the contrast is  stretching  The formula requires finding the minimum and maximum pixel intensity multiply by levels of gray. In our case the image is 8bpp, so levels of gray are 256.  The minimum value is 0 and the maximum value is 225. So the formula in our case is  stretching  where f(x,y) denotes the value of each pixel intensity. For each f(x,y) in an image , we will calculate this formula.  After doing this, we will be able to enhance our contrast.  The following image appear after applying histogram stretching.  stretching  The stretched histogram of this image has been shown below.  Note the shape and symmetry of histogram. The histogram is now stretched or in other means expand. Have a look at it.  stretching  In this case the contrast of the image can be calculated as  Contrast = 240  Hence we can say that the contrast of the image is increased.  **Note** : this method of increasing contrast doesnot work always, but it fails on some cases.  Failing of histogram stretching  As we have discussed , that the algorithm fails on some cases. Those cases include images with when there is pixel intensity 0 and 255 are present in the image  Because when pixel intensities 0 and 255 are present in an image, then in that case they become the minimum and maximum pixel intensity which ruins the formula like this.  Original Formula  stretching  Putting fail case values in the formula:  stretching  Simplify that expression gives  stretching  That means the output image is equal to the processed image. That means there is no effect of histogram stretching has been done at this image.  **Spatial Filtering** technique is used directly on pixels of an image. Mask is usually considered to be added in size so that it has specific center pixel. This mask is moved on the image such that the center of the mask traverses all image pixels.  **Classification on the basis of linearity:** There are two types:  **1.** Linear Spatial Filter  **2.** Non-linear Spatial Filter  **General Classification:**  **Smoothing Spatial Filter:** Smoothing filter is used for blurring and noise reduction in the image. Blurring is pre-processing steps for removal of small details and Noise Reduction is accomplished by blurring.  **Types of Smoothing Spatial Filter:**  **1.** Linear Filter (Mean Filter)  **2.** Order Statistics (Non-linear) filter  These are explained as following below.   1. **Mean Filter:** Linear spatial filter is simply the average of the pixels contained in the neighborhood of the filter mask. The idea is replacing the value of every pixel in an image by the average of the grey levels in the neighborhood define by the filter mask.   **Types of Mean filter:**   * + **(i) Averaging filter:** It is used in reduction of the detail in image. All coefficients are equal.   + **(ii) Weighted averaging filter:** In this, pixels are multiplied by different coefficients. Center pixel is multiplied by a higher value than average filter.  1. **Order Statistics Filter:** It is based on the ordering the pixels contained in the image area encompassed by the filter. It replaces the value of the center pixel with the value determined by the ranking result. Edges are better preserved in this filtering.   **Types of Order statistics filter:**   * + **(i) Minimum filter:** 0th percentile filter is the minimum filter. The value of the center is replaced by the smallest value in the window.   + **(ii) Maximum filter:** 100th percentile filter is the maximum filter. The value of the center is replaced by the largest value in the window.   + **(iii) Median filter:** Each pixel in the image is considered. First neighboring pixels are sorted and original values of the pixel is replaced by the median of the list.   **Sharpening Spatial Filter:** It is also known as derivative filter. The purpose of the sharpening spatial filter is just the opposite of the smoothing spatial filter. Its main focus in on the removal of blurring and highlight the edges. It is based on the first and second order derivative.  **First order derivative:**   * Must be zero in flat segments. * Must be non zero at the onset of a grey level step. * Must be non zero along ramps.   First order derivative in 1-D is given by:  f' = f(x+1) - f(x)  **Second order derivative:**   * Must be zero in flat areas. * Must be zero at the onset and end of a ramp. * Must be zero along ramps.   Second order derivative in 1-D is given by:  f'' = f(x+1) + f(x-1) - 2f(x)  **Frequency Domain Filters** are used for smoothing and sharpening of image by removal of high or low frequency components. Sometimes it is possible of removal of very high and very low frequency. Frequency domain filters are different from spatial domain filters as it basically focuses on the frequency of the images. It is basically done for two basic operation i.e., Smoothing and Sharpening.  These are of 3 types:     1. **Low pass filter:** Low pass filter removes the high frequency components that means it keeps low frequency components. It is used for smoothing the image. It is used to smoothen the image by attenuating high frequency components and preserving low frequency components. The mechanism of low pass filtering in the frequency domain is given by:   G(u, v) = H(u, v) . F(u, v)  where F(u, v) is the Fourier Transform of original image  and H(u, v) is the Fourier Transform of filtering mask  **2. High pass filter:** High pass filter removes the low frequency components that means it keeps high frequency components. It is used for sharpening the image. It is used to sharpen the image by attenuating low frequency components and preserving high frequency components. Mechanism of high pass filtering in frequency domain is given by:  H(u, v) = 1 - H'(u, v)  where H(u, v) is the Fourier Transform of high pass filtering  and H'(u, v) is the Fourier Transform of low pass filtering  **3. Band pass filter:** Band pass filter removes the very low frequency and very high frequency components that means it keeps the moderate range band of frequencies. Band pass filtering is used to enhance edges while reducing the noise at the same time. |

|  |
| --- |
| **Stepwise-Procedure:** |
| 1. Click Selfie and transfer the image to the PC 2. Apply histogram equalization and compare it with image stretching 3. Apply Low pass filter in spatial and frequency domain 4. Apply High pass filter in spatial and frequency domain 5. Prove high boost filter is the same as the 1-low pass filter if A is 1 6. Perform the same experiment in Matlab/python |

|  |
| --- |
| **Output** |
| Upload picture screenshots for all approaches with intermediate steps  **HISTOGRAM STRETCHING**  clear all;  close all;  clc;  selfie = imread(vedant.jpg');  graySelfie = rgb2gray(selfie);  [r,c] = size(graySelfie);  figure;  imshow(graySelfie);  title('gray Selfie');  figure;  imhist(graySelfie, 50);  title('Histogram of gray Selfie');  % we can observe maximum pixels are in range 180 to 220  % max value seems to be around 200  stretchedSelfie = zeros(r,c);  for i=1:r  for j=1:c  stretchedSelfie(i,j) = graySelfie(i,j);  end  end  for i=1:r  for j=1:c    stretchedSelfie(i,j) = (graySelfie(i,j)-150)/(200-150)\*100;  end  end  stretchedSelfie = uint8(stretchedSelfie);  figure;  imshow(stretchedSelfie);  title('gray stretched Selfie');  figure;  imhist(stretchedSelfie, 50);  title('Histogram of gray stretched Selfie');          **FREQUENCY DOMAIN FILTERING**  clear all;  close all;  clc;  img = imread('manush.jpg');  figure;  imshow(img);  title('Original Image');  gray=rgb2gray(img);  figure;  imshow(gray);  title('gray image');  windowSize = 10;  kernel = ones(windowSize, windowSize) / windowSize ^ 2;  avg\_spatial = imfilter(gray, kernel, 'symmetric');  figure;  imshow(avg\_spatial);  title('Average filter');  %Frequency response  F=fft2(gray);  S=fftshift(log(1+abs(F)));  figure;  imshow(S,[]);  [M, N] = size(gray);  FT\_img = fft2(double(gray));  n = 40; % filter order  D0 = 50; % cutoff frequency    % Designing filter  u = 0:(M-1);  v = 0:(N-1);  idx = find(u > M/2);  u(idx) = u(idx) - M;  idy = find(v > N/2);  v(idy) = v(idy) - N;    % MATLAB library function meshgrid(v, u) returns  % 2D grid which contains the coordinates of vectors  % v and u. Matrix V with each row is a copy of v  % and matrix U with each column is a copy of u  [V, U] = meshgrid(v, u);    % Calculating Euclidean Distance  D = sqrt(U.^2 + V.^2);  H = 1./(1 + (D./D0).^(2\*n)); % filter mask    % Convolution between the Fourier image and mask  G = H.\*FT\_img;    % Getting the resultant image by Inverse Fourier Transform  % of the convoluted image using MATLAB library function  % ifft2 (2D inverse fast fourier transform)  output\_image = real(ifft2(double(G)));    % Displaying Input Image and Output Image  figure;  imshow(output\_image,[]);  % subplot(2, 1, 1), imshow(gray),  % subplot(2, 1, 2), imshow(output\_image, [ ]); |

|  |
| --- |
| **Conclusions:** |
| In this experimenr, we preform histogram stretching and frequency domain average filtering on input image. |

|  |
| --- |
| **Post Lab Subjective/Objective type Questions:** |
| Answer the following questions:   1. What are different operations that can be performed using histogram  ANS: Contrast stretching-Frequently, an image is scanned in such a way that the resulting brightness values do not make full use of the available dynamic range. This can be easily observed in the histogram of the brightness values shown in Figure 6. By stretching the histogram over the available dynamic range we attempt to correct this situation. If the image is intended to go from brightness 0 to brightness 2B-1 (see Section 2.1), then one generally maps the 0% value (or minimum as defined in Section 3.5.2) to the value 0 and the 100% value (or maximum) to the value 2B-1Equalization-ANS: The most common histogram normalization technique is histogram equalization where one attempts to change the histogram through the use of a function b = http://vidya.amrita.ac.in/electronics/image-process/www.ph.tn.tudelft.nl/Courses/FIP/images/symbol/florin.gif(a) into a histogram that is constant for all brightness values. This would correspond to a brightness distribution where all values are equally probable. The histogram derived from a local region can also be used to drive local filters that are to be applied to that region.Examples include minimum filtering, median filtering, and maximum filtering.  1. What is the main difference between filtering at the Frequency domain and Spatial domain   **ANS**: In theory, convolution in the time domain is the same as multiplication in the frequency domain. And filtering a signal with a linear filter is mathematically the same as convolving by its impulse response. So, theoretically there is no difference. Practically, doing it in the frequency domain is slightly more complicated, but it can be faster. For a large enough filter kernel, using the FFT is usually a lot faster than filtering in time domain-- particularly because you can use a technique called "overlap-add" to speed things up. |

|  |
| --- |
| **Signature of faculty in-charge with Date:** |